

**DRAFT – NOT TO BE ENTERED****Application Serial No. : 09/583,672****Proposed claim amendments:**

1. (Currently Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for implementing an object type-declaration syntax, comprising:

allowing a type declaration in a programming language to be embedded within an object identifier declaration; and

allowing the type declaration to be delimited from the object identifier declaration using a joint attribute; and

allowing the object identifier having the type declaration to be used throughout a life of a program as a syntax for referencing an object in the program.

16. (Currently Amended) A method of declaring an object type in a programming language, comprising:

embedding an object type indicator with an object identifier name, wherein the object identifier name is interpreted by a machine as having the object type indicator and is used throughout a life of a program as a syntax for referencing an object in the program.

17. (Currently Amended) A method of declaring an object type in a programming language, comprising:

prepending an object type indicator with an object identifier name, wherein the object identifier name is interpreted by a machine as having the object type indicator and is used throughout a life of a program as a syntax for referencing an object in the program.

20. (Currently Amended) The program storage device of claim 1, wherein the type declaration allows a compiler or interpreter of the programming language to operate on an object declared in the type declaration without an explicit call to construct the object.

21. (Currently Amended) The program storage device of claim 1, wherein the type declaration allows a compiler or interpreter of the programming language to automatically instantiate an object being declared in the type declaration when the type declaration embedded with the object identifier declaration is first read by the programming language compiler or interpreter.

**Brief Description of arguments to be presented:**

Bernes-Lee does not discuss **programming language syntax**. If Bernes-Lee protocol is used in a programming language “throughout a life of a program as a syntax for referencing an object in the program,” program compilers or interpreters would not know how to handle it. For example, one cannot assign a value to the Bernes-Lee’s protocol expression.

In Staugaard and JLE, C++ syntax “char x” is only used once for explicitly declaring that x will hold one character type. Once the “char” type is declared, the variable expression “x” is used alone without its explicit type throughout the program. The type declaring in Staugaard and JLE cannot be used more than once in a programming. For example, one cannot declare “char x='a” then use “char x='b” again in the same program. Further, one cannot say char x='a', then use int x=1.